JUSTE PRIX

|  |
| --- |
| <!DOCTYPE html> |
|  | <html lang="en"> |
|  | <head> |
|  | <meta charset="UTF-8"> |
|  | <title>Maths</title> |
|  | </head> |
|  | <body> |
|  |  |
|  | <p>Regarde la console</p> |
|  |  |
|  |  |
|  | <script> |
|  |  |
|  | /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* |
|  | LE JUSTE PRIX |
|  | \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/ |
|  |  |
|  | let message = ""; |
|  | let trials = 0; |
|  | let min\_value = 20; |
|  | let max\_value = 80; |
|  |  |
|  | // fonction qui retourne un numéro aléatoire arondi entre la variable minimale et la variable maximale |
|  | let random\_range = function(min, max){ |
|  | return Math.round( Math.random() \* (max - min) + min ); |
|  | } |
|  |  |
|  | let price = random\_range(min\_value, max\_value); |
|  |  |
|  | console.log("Bienvenu au juste prix !"); |
|  | console.log("Le prix de mon drone va de " + min\_value + " à " + max\_value + " euros"); |
|  | console.log("Écris dans la console \"guessNumber(20)\" pour demander si le prix est de 20 euros."); |
|  |  |
|  | let guessNumber = function(number){ |
|  |  |
|  | trials = trials + 1; |
|  |  |
|  | if(number < price){ |
|  | message = "C'est plus"; |
|  | } else if(number > price){ |
|  | message = "C'est moins"; |
|  | } else { |
|  | message = "C'est juste tu as trouvé en " + trials + " coups"; |
|  | } |
|  | return message; |
|  | } |
|  |  |
|  | </script> |
|  | </body> |
|  | </html> |

OBJET

|  |
| --- |
| <!DOCTYPE html> |
|  | <html lang="en"> |
|  | <head> |
|  | <meta charset="UTF-8"> |
|  | <title>Maths</title> |
|  | </head> |
|  | <body> |
|  |  |
|  | <p>Regarde la console</p> |
|  |  |
|  | <script> |
|  |  |
|  | /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* |
|  | EXERCICE 1 |
|  | \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/ |
|  |  |
|  | let character = { |
|  | name : "Teddy", |
|  | age : 29, |
|  | items\_to\_give : ["Épée","couteau","pierre"], |
|  | giveItem : function(){ |
|  | let key = Math.floor(Math.random() \* this.items\_to\_give.length); |
|  | return this.items\_to\_give[key]; |
|  | } |
|  | } |
|  |  |
|  | for(let key in character){ |
|  | console.log( key + " : " + character[key] ); |
|  | } |
|  |  |
|  | let object\_received = character.giveItem(); |
|  |  |
|  | console.log(character.name + " vous donne " + object\_received ); |
|  |  |
|  | /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* |
|  | EXERCICE 2 |
|  | \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/ |
|  |  |
|  | let shop = [ |
|  | { title : "épée", physic : 2, magic : 5, minLevel : 1, available : true }, |
|  | { title : "hache", physic : 3, magic : 10, minLevel : 3, available : false }, |
|  | { title : "sceptre", physic : 4, magic : 15, minLevel : 6, available : true } |
|  | ] |
|  |  |
|  | let display\_objects = function(){ |
|  | for (var i = 0; i < shop.length; i++) { |
|  | console.log( shop[i] ); |
|  | } |
|  | } |
|  |  |
|  | let display\_available\_objects = function(){ |
|  | for (var i = 0; i < shop.length; i++) { |
|  | if(shop[i].available){ |
|  | console.log( shop[i] ); |
|  | } |
|  | } |
|  | } |
|  |  |
|  | let display\_level\_objects = function(){ |
|  | for (var i = 0; i < shop.length; i++) { |
|  | if(shop[i].minLevel <= 10){ |
|  | console.log( shop[i] ); |
|  | } |
|  | } |
|  | } |
|  |  |
|  | console.log("----display\_objects----"); |
|  | display\_objects(); |
|  |  |
|  | console.log("----display\_available\_objects----"); |
|  | display\_available\_objects(); |
|  |  |
|  | console.log("----display\_level\_objects----"); |
|  | display\_level\_objects(); |
|  |  |
|  |  |
|  | /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* |
|  | EXERCICE 3 |
|  | \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/ |
|  |  |
|  | let mainCharacter = { |
|  | name : "Barret", |
|  | level : 10, |
|  | life : 100, |
|  | weapon : { |
|  | name : "Hand Gun", |
|  | damage : 3 |
|  | }, |
|  | attack : function(){ |
|  | let total\_damage = this.level \* this.weapon.damage; |
|  | return this.name + " attaque avec l'arme " + this.weapon.name + " les dégâts sont " + total\_damage + " points."; |
|  | } |
|  | } |
|  |  |
|  | console.log( mainCharacter.attack() ); |
|  |  |
|  | /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* |
|  | EXERCICE 4 : Bonus |
|  | \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/ |
|  |  |
|  | let Character = { |
|  | name : "", |
|  | level : 1, |
|  | life : 100, |
|  | weapon : { |
|  | name : "Épée", |
|  | damage : 3 |
|  | }, |
|  | attack : function(opponent){ |
|  |  |
|  | let total\_damage = this.level \* this.weapon.damage; |
|  |  |
|  | opponent.receiveDamage( total\_damage ); |
|  |  |
|  | console.log( this.name + " attaque " + opponent.name + " avec l'arme " + this.weapon.name + "." ); |
|  | console.log("Il lui inflige " + total\_damage + " points de dégats."); |
|  | console.log(opponent.name + " a maintenant " + opponent.life + " de vie"); |
|  |  |
|  | }, |
|  | receiveDamage : function(damage){ |
|  |  |
|  | this.life = this.life - damage; |
|  |  |
|  | } |
|  | } |
|  |  |
|  | let heroCharacter = Object.create(Character); |
|  | let opponentCharacter = Object.create(Character); |
|  |  |
|  | heroCharacter.name = "Séphiroth"; |
|  | heroCharacter.level = 5; |
|  |  |
|  | opponentCharacter.name = "Cloud"; |
|  | opponentCharacter.level = 3; |
|  |  |
|  | heroCharacter.attack(opponentCharacter); |
|  |  |
|  | console.log( heroCharacter ); |
|  | console.log( opponentCharacter ); |
|  |  |
|  |  |
|  | </script> |
|  | </body> |
|  | </html> |

JUSTE PRIX

|  |
| --- |
| <!DOCTYPE html> |
|  | <html lang="en"> |
|  | <head> |
|  | <meta charset="UTF-8"> |
|  | <title>Maths</title> |
|  | </head> |
|  | <body> |
|  |  |
|  | <p>Regarde la console</p> |
|  |  |
|  |  |
|  | <script> |
|  |  |
|  | /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* |
|  | LE JUSTE PRIX |
|  | \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/ |
|  |  |
|  | let message = ""; |
|  | let trials = 0; |
|  | let min\_value = 20; |
|  | let max\_value = 80; |
|  |  |
|  | // fonction qui retourne un numéro aléatoire arondi entre la variable minimale et la variable maximale |
|  | let random\_range = function(min, max){ |
|  | return Math.round( Math.random() \* (max - min) + min ); |
|  | } |
|  |  |
|  | let price = random\_range(min\_value, max\_value); |
|  |  |
|  | console.log("Bienvenu au juste prix !"); |
|  | console.log("Le prix de mon drone va de " + min\_value + " à " + max\_value + " euros"); |
|  | console.log("Écris dans la console \"guessNumber(20)\" pour demander si le prix est de 20 euros."); |
|  |  |
|  | let guessNumber = function(number){ |
|  |  |
|  | trials = trials + 1; |
|  |  |
|  | if(number < price){ |
|  | message = "C'est plus"; |
|  | } else if(number > price){ |
|  | message = "C'est moins"; |
|  | } else { |
|  | message = "C'est juste tu as trouvé en " + trials + " coups"; |
|  | } |
|  | return message; |
|  | } |
|  |  |
|  | </script> |
|  | </body> |
|  | </html> |

DOM

|  |
| --- |
| <!DOCTYPE html> |
|  | <html lang="en"> |
|  | <head> |
|  | <meta charset="UTF-8"> |
|  | <title>Maths</title> |
|  | </head> |
|  | <body> |
|  |  |
|  | <p>Regarde la console</p> |
|  |  |
|  |  |
|  | <script> |
|  |  |
|  | /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* |
|  | LE JUSTE PRIX |
|  | \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/ |
|  |  |
|  | let message = ""; |
|  | let trials = 0; |
|  | let min\_value = 20; |
|  | let max\_value = 80; |
|  |  |
|  | // fonction qui retourne un numéro aléatoire arondi entre la variable minimale et la variable maximale |
|  | let random\_range = function(min, max){ |
|  | return Math.round( Math.random() \* (max - min) + min ); |
|  | } |
|  |  |
|  | let price = random\_range(min\_value, max\_value); |
|  |  |
|  | console.log("Bienvenu au juste prix !"); |
|  | console.log("Le prix de mon drone va de " + min\_value + " à " + max\_value + " euros"); |
|  | console.log("Écris dans la console \"guessNumber(20)\" pour demander si le prix est de 20 euros."); |
|  |  |
|  | let guessNumber = function(number){ |
|  |  |
|  | trials = trials + 1; |
|  |  |
|  | if(number < price){ |
|  | message = "C'est plus"; |
|  | } else if(number > price){ |
|  | message = "C'est moins"; |
|  | } else { |
|  | message = "C'est juste tu as trouvé en " + trials + " coups"; |
|  | } |
|  | return message; |
|  | } |
|  |  |
|  | </script> |
|  | </body> |
|  | </html> |

AJAX

|  |
| --- |
| <!DOCTYPE html> |
|  | <html lang="en"> |
|  | <head> |
|  | <meta charset="UTF-8"> |
|  | <title>Ajax</title> |
|  |  |
|  | </head> |
|  | <body> |
|  |  |
|  | <p>Regarde la console</p> |
|  | <p>Pour que l'exemple fonctionne, il faut l'executer soit sur Firefox, soit sur un serveur.</p> |
|  |  |
|  | <script> |
|  | // déclare qu'on va faire une requête |
|  | let request = new XMLHttpRequest(); |
|  | /\* |
|  | prépare la requête |
|  | query.open(méthode ?, url ?, asyncrone ?); |
|  | \*/ |
|  | request.open('GET', 'data.json', true); |
|  | // j'envoie la requête |
|  | request.send(); |
|  | // quand la requête est finie on execute cette fonction |
|  | request.onload = function() { |
|  | // vérifie si tout s'est bien passé |
|  | if (request.status >= 200 && request.status < 400) { |
|  | // je recois le JSON sous forme de texte |
|  | let response = request.responseText; |
|  | // je transforme le texte en JSON |
|  | let reponse\_to\_JSON = JSON.parse(response); |
|  |  |
|  | /\* |
|  | pour éviter d'avoir plein de code dans mon 'if' |
|  | - execute le fonction 'manipulate\_data' |
|  | - je donne 'reponse\_to\_JSON' en argument |
|  | \*/ |
|  | manipulate\_data(reponse\_to\_JSON); |
|  |  |
|  | } else { |
|  | console.log("l'URL a bien été atteinte mais elle a retourné un érreur"); |
|  | } |
|  | }; |
|  |  |
|  | // si il y a eu une érreur pendant l'envoie de la requête |
|  | request.onerror = function() { |
|  | console.log("Il y a eu une érreur quelque part..."); |
|  | }; |
|  |  |
|  | // re reçois mon json en argument |
|  | let manipulate\_data = function(json\_data){ |
|  |  |
|  | // ajouter un score aléatoire (0 -> 1000) à toutes ces personnes |
|  | json\_data.map(function(person, key){ |
|  | person.score = Math.round( Math.random() \* 1000 ); |
|  | }) |
|  |  |
|  | // classer l'odre du tableau selon les scores |
|  | json\_data.sort(function(personA, personB){ |
|  | return personA.score - personB.score; |
|  | }) |
|  |  |
|  | // création des 3 categories |
|  | let category\_a = []; |
|  | let category\_b = []; |
|  | let category\_c = []; |
|  |  |
|  | // pour chaque personne (boucle) je mets la pousse dans le bon tableau selon son score |
|  | json\_data.forEach(function(person, key){ |
|  | if(person.score > 750){ |
|  | category\_a.push(person); |
|  | } else if( person.score > 500 ){ |
|  | category\_b.push(person); |
|  | } else { |
|  | category\_c.push(person); |
|  | } |
|  | }) |
|  |  |
|  | console.log("Category A"); |
|  | console.log(category\_a); |
|  | console.log("Category B"); |
|  | console.log(category\_b); |
|  | console.log("Category C"); |
|  | console.log(category\_c); |
|  |  |
|  | // crécupère les personne qui viennent du "Bahrain" |
|  | let bahrain\_people = json\_data.filter(function(person, key){ |
|  | if (person.country == "Bahrain") { |
|  | return true; |
|  | } else { |
|  | return false; |
|  | } |
|  | }) |
|  |  |
|  | // compter le nombre de personne qu'il y a dans le tableau 'bahrain\_people' |
|  | let bahrain\_people\_count = bahrain\_people.length; |
|  |  |
|  | console.log("bahrain people count : " + bahrain\_people\_count); |
|  |  |
|  | // Ayant déjà utilisé 'array.sort' pour classer les personnes par score |
|  | // le plus grand score |
|  | let person\_with\_highest\_score = json\_data[json\_data.length - 1]; |
|  | let highest\_score = person\_with\_highest\_score.score; |
|  |  |
|  | // le plus petit score |
|  | let person\_with\_lowest\_score = json\_data[0]; |
|  | let lowest\_score = person\_with\_lowest\_score.score; |
|  |  |
|  | console.log("highest score : " + highest\_score); |
|  | console.log("lowest score : " + lowest\_score); |
|  |  |
|  | } |
|  |  |
|  |  |
|  | </script> |
|  | </body> |
|  | </html> |

MOUSE-EVENT

|  |
| --- |
| <!DOCTYPE html> |
|  | <html lang="en"> |
|  | <head> |
|  | <meta charset="UTF-8"> |
|  | <title>Mouse</title> |
|  | <style> |
|  | .square-red { |
|  | width: 100px; |
|  | height: 100px; |
|  | background: red; |
|  | margin: 20px; |
|  | opacity: 1; |
|  | transition: opacity 0.5s; |
|  | } |
|  | .hide { |
|  | opacity: 0; |
|  | } |
|  | </style> |
|  | </head> |
|  | <body> |
|  |  |
|  | <div id="reset">Reset</div> |
|  |  |
|  | <div class="square-red"></div> |
|  | <div class="square-red"></div> |
|  | <div class="square-red"></div> |
|  |  |
|  | <div id="x-axis"></div> |
|  | <div id="y-axis"></div> |
|  |  |
|  |  |
|  |  |
|  | <script> |
|  |  |
|  | // Évènements de la souris |
|  | // click, ctextmenu, dblclick, mousedown, mouseenter, mouseleave, mousemove, mouseover, mouseout, mouseup |
|  |  |
|  | /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* |
|  | EXERCICE 1 |
|  | \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/ |
|  |  |
|  | let square\_red\_elements = document.querySelectorAll(".square-red"); |
|  |  |
|  | var mouseEntered\_square\_Red = function(event){ |
|  | let target\_element = event.target; |
|  | target\_element.classList.add("hide"); |
|  | } |
|  | square\_red\_elements.forEach(function(square\_red\_element){ |
|  | square\_red\_element.addEventListener("mouseenter", mouseEntered\_square\_Red); |
|  | }) |
|  |  |
|  | /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* |
|  | EXERCICE 2 |
|  | \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/ |
|  |  |
|  | let reset\_element = document.querySelector("#reset"); |
|  |  |
|  | let display\_square\_red\_elements = function(){ |
|  | square\_red\_elements.forEach(function(square\_red\_element){ |
|  | square\_red\_element.classList.remove("hide"); |
|  | }) |
|  | } |
|  |  |
|  | reset\_element.addEventListener("click", display\_square\_red\_elements); |
|  |  |
|  |  |
|  | /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* |
|  | EXERCICE 3 |
|  | \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/ |
|  |  |
|  | let xAxis\_element = document.querySelector("#x-axis"); |
|  | let yAxis\_element = document.querySelector("#y-axis"); |
|  |  |
|  | let displayCooDinates = function(event){ |
|  | xAxis\_element.innerText = "X : " + event.clientX; |
|  | yAxis\_element.innerText = "Y : " + event.clientY; |
|  | } |
|  |  |
|  | window.addEventListener("mousemove", displayCooDinates); |
|  |  |
|  |  |
|  |  |
|  | </script> |
|  | </body> |
|  | </html> |

KEYBOARD-EVENT

|  |
| --- |
| <!DOCTYPE html> |
|  | <html lang="en"> |
|  | <head> |
|  | <meta charset="UTF-8"> |
|  | <title>Maths</title> |
|  |  |
|  | <style> |
|  | #character { |
|  | width: 100px; |
|  | height: 100px; |
|  | background: red; |
|  | margin: 20px; |
|  | text-align: center; |
|  | line-height: 100px; |
|  | } |
|  | .arrows { |
|  | width: 100px; |
|  | height: 100px; |
|  | background: blue; |
|  | margin: 20px; |
|  | float: left; |
|  | text-align: center; |
|  | line-height: 100px; |
|  | } |
|  | .highlight { |
|  | background: yellow; |
|  | } |
|  | .rounded { |
|  | border-radius: 50%; |
|  | } |
|  | </style> |
|  | </head> |
|  | <body> |
|  |  |
|  | <div id="character">0 - 10</div> |
|  |  |
|  | <br> |
|  |  |
|  | <div id="up" class="arrows">Up</div> |
|  | <div id="down" class="arrows">Down</div> |
|  | <div id="left" class="arrows">Left</div> |
|  | <div id="right" class="arrows">Right</div> |
|  |  |
|  |  |
|  | <script> |
|  |  |
|  | /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* |
|  | EXERCICE 1 |
|  | \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/ |
|  |  |
|  | let character = document.querySelector("#character"); |
|  |  |
|  | let change\_character\_color = function(event){ |
|  | let bg\_color = ""; |
|  | switch(event.key){ |
|  | case "0" : bg\_color = "red"; break; |
|  | case "1" : bg\_color = "blue"; break; |
|  | case "2" : bg\_color = "green"; break; |
|  | case "3" : bg\_color = "cyan"; break; |
|  | case "4" : bg\_color = "magenta"; break; |
|  | case "5" : bg\_color = "yellow"; break; |
|  | case "6" : bg\_color = "orange"; break; |
|  | case "7" : bg\_color = "purple"; break; |
|  | case "8" : bg\_color = "olive"; break; |
|  | case "9" : bg\_color = "black"; break; |
|  | } |
|  | character.style.background = bg\_color; |
|  | } |
|  |  |
|  | window.addEventListener("keyup", change\_character\_color); |
|  |  |
|  |  |
|  | /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* |
|  | EXERCICE 2 |
|  | \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/ |
|  |  |
|  | let up = document.querySelector("#up"); |
|  | let down = document.querySelector("#down"); |
|  | let left = document.querySelector("#left"); |
|  | let right = document.querySelector("#right"); |
|  |  |
|  | let show\_arrow\_key = function(event){ |
|  |  |
|  | switch(event.key){ |
|  | case "ArrowUp" : |
|  | up.classList.add("highlight"); |
|  | break; |
|  | case "ArrowDown" : |
|  | down.classList.add("highlight"); |
|  | break; |
|  | case "ArrowLeft" : |
|  | left.classList.add("highlight"); |
|  | break; |
|  | case "ArrowRight" : |
|  | right.classList.add("highlight"); |
|  | break; |
|  | } |
|  | } |
|  | let hide\_arrow\_key = function(event){ |
|  |  |
|  | switch(event.key){ |
|  | case "ArrowUp" : |
|  | up.classList.remove("highlight"); |
|  | break; |
|  | case "ArrowDown" : |
|  | down.classList.remove("highlight"); |
|  | break; |
|  | case "ArrowLeft" : |
|  | left.classList.remove("highlight"); |
|  | break; |
|  | case "ArrowRight" : |
|  | right.classList.remove("highlight"); |
|  | break; |
|  | } |
|  | } |
|  |  |
|  | window.addEventListener("keydown", show\_arrow\_key); |
|  | window.addEventListener("keyup", hide\_arrow\_key); |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  | </script> |
|  | </body> |
|  | </html> |